ABSTRACT

A stateless distributed computer architecture allows state-caching objects, which hold server state information, to be maintained on a client or network rather than on a server. In one implementation, the computer architecture implements object-oriented program modules according to a distributed component object model (DCOM). Using an object-oriented network protocol (e.g., remote procedure call), a client-side application calls through an application program interface (API) to a program object residing at a server computer. The program object, responsive to the call, creates a state caching object that contains state information pertaining to the client connection. The server inserts the state-caching object into a local thread context and processes the request to generate a reply. The server subsequently attaches the state-caching object to the reply and returns them both to the client. The client stores the state-caching object for later communication with the server. When the client subsequently calls the program object at the server, the client submits the state-caching object along with the request packet. The server uses the state information in the state-caching object to quickly restore state for the client reconnection. In this manner, the server can offload its state information to other computing devices in the distributed architecture, thereby improving scalability. In another implementation, the network itself caches the server-oriented state-caching object.